

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1849	715/513	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L2	2	"6687707".pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L3	2	"6292827".pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L4	2	"5410646".pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L5	613	L1 and attributes and identify and input	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L6	590	L1 and attributes and identify and input and user	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L7	2	"5704029".pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L8	2	"5842195".pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21

L9	2	"5893098".pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L10	2	"6070143".pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L11	2	"6189029".pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L12	2	"6151581".pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L13	5500	attributes and value and input and user and html and xml and form	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L14	383	attributes and value and input and user and html and xml and form and 715/513	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L15	371	attributes and value and input and user and html and xml and form and 715/513 and field	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L16	236	715/505	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21

L17	116	715/506	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L18	208	715/507	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21
L19	71	715/508	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/01/12 16:21


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide


[Feedback](#) [Report](#)

 Terms used attributes and program and form and XML

 Sort results by 

 Display results 
☒ [Save results to a Binder](#)
☐ [Search Tips](#)
☐ [Open results in a new window](#)

 Try an /  
Try this

Results 1 - 20 of 200

 Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

### 1 [Extending Java for high-level Web service construction](#)



Aske Simon Christensen, Anders Møller, Michael I. Schwartzbach

 November 2003 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volu

Publisher: ACM Press

 Full text available: [pdf\(947.02 KB\)](#)

 Additional Information: [full citation](#), [abstract](#), [references](#), [cit](#)

We incorporate innovations from the <bigwig> project into the Java language to provide high-le. The resulting language, Jwig, contains an advanced session model and a flexible mechanism for particular XHTML. To support program development we provide a suite of program analyses that that no runtime errors can occur while building documents or receiving form input, and ...

**Keywords:** Interactive Web services, XML, data-flow analysis

### 2 [Toward an engineering discipline for grammarware](#)



Paul Klint, Ralf Lämmel, Chris Verhoef

 July 2005 **ACM Transactions on Software Engineering and Methodology (TOSEM)**, Volu

Publisher: ACM Press

 Full text available: [pdf\(710.42 KB\)](#)

 Additional Information: [full citation](#), [abstract](#), [references](#), [in](#)

Grammarware comprises grammars and all grammar-dependent software. The term *grammar* is grammar formalisms and grammar notations including context-free grammars, class dictionaries tree and graph grammars. The term *grammar-dependent software* refers to all software that inv manner. Archetypal examples of grammar-dependent software are parsers, program converters

**Keywords:** Grammarware, automated software engineering, best practices, generic language te grammars, language processing, metamodeling, model-driven development, parsers, software e

### 3 [Haskell and XML: generic combinators or type-based translation?](#)



Malcolm Wallace, Colin Runciman

 September 1999 **ACM SIGPLAN Notices , Proceedings of the fourth ACM SIGPLAN internatio programming ICFP '99**, Volume 34 Issue 9

Publisher: ACM Press

 Full text available: [pdf\(1.48 MB\)](#)

 Additional Information: [full citation](#), [abstract](#), [references](#), [cit](#)

We present two complementary approaches to writing XML document-processing applications in the generic tree structure of XML documents is used as the basis for the design of a library of co generation, and transformation of XML trees. The second approach is to use a type-translation fr definitions (DTDs) as declarations of algebraic data types, and a derivation of the cor ...

#### 4 Posters: Consistency checking of UML model diagrams using the XML semantics approach



Yasser Kotb, Takuya Katayama

May 2005

**Special interest tracks and posters of the 14th international conference on**

**Publisher:** ACM Press

Full text available: [pdf\(268.05 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [in](#)

A software design is often modeled as a collection of unified Modeling Language (UML) diagrams system that are covered by many different UML diagrams. This leads for big risk that the overall inconsistent and incompleteness. This inherits the necessary to check the consistency between the software system gets evolution, those diagrams get modified that leads again to possi ...

**Keywords:** UML, XMI, XML, attribute grammars, model checking

#### 5 Anatomy of a native XML base management system

T. Fiebig, S. Helmer, C.-C. Kanne, G. Moerkotte, J. Neumann, R. Schiele, T. Westmann

December 2002 **The VLDB Journal — The International Journal on Very Large Data Bases**, v

**Publisher:** Springer-Verlag New York, Inc.

Full text available: [pdf\(300.97 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [citations](#), [index](#)

Several alternatives to manage large XML document collections exist, ranging from file systems to specifically tailored XML base management systems. In this paper we give a tour of Natix, a data scratch for storing and processing XML data. Contrary to the common belief that management of traditional databases like relational systems, we illustrate how almost every component in a ...

**Keywords:** Database, XML

#### 6 Transformations and Experiences: Towards static type checking for XSLT



Akihiko Tozawa

November 2001

**Proceedings of the 2001 ACM Symposium on Document engineering**

**Publisher:** ACM Press

Full text available: [pdf\(175.21 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [cit](#)

We are concerned about the *static type checking* problem for XSLT. In the context of XSLT and c schemas, and *static type checking* is to verify that a program always converts valid source docum achieve static type checking for XSLT, we introduce a subset of XSLT, and an efficient algorithm Although our XSLT subset lacks XPath, it includes recursiv ...

#### 7 Concrete syntax for objects: domain-specific language embedding and assimilation without



Martin Bravenboer, Eelco Visser

October 2004

**ACM SIGPLAN Notices , Proceedings of the 19th annual ACM SIGPLAN conf systems, languages, and applications OOPSLA '04**, Volume 39 Issue 10

**Publisher:** ACM Press

Full text available: [pdf\(379.91 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [in](#)

Application programmer's interfaces give access to domain knowledge encapsulated in class libra notation for expressing domain composition. Since object-oriented languages are designed for ex constructs are often sufficient for expressing domain abstractions at the semantic level. However at the syntactic level. In this paper we describe MetaBorg, a method for providing <i> ...

**Keywords:** MetaBorg, SDF, concrete object syntax, domain-specific languages, embedded langu rewriting, stratego, syntax extension

8 Change management: An infrastructure for development of object-oriented, multi-level con

Tien N. Nguyen, Ethan V. Munson, John T. Boyland, Cheng Thao

May 2005

**Proceedings of the 27th international conference on Software engineering****Publisher:** ACM Press

Full text available: pdf(418.74 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [info](#)

In an integrated development environment, the ability to manage the evolution of a software system compositions, and their interrelations is crucial to successful software development. This paper presents an infrastructure, *Molhado*, upon which to build *object-oriented* software configuration management integrated development environment. Key contributions of this paper include a *product versioning*.

**Keywords:** *software configuration management, version control*

9 XML parsing and stylesheets: An adaptive, fast, and safe XML parser based on byte sequence

Toshiro Takase, Hisashi MIYASHITA, Toyotaro Suzumura, Michiaki Tatsubori

May 2005

**Proceedings of the 14th international conference on World Wide Web****Publisher:** ACM Press

Full text available: pdf(274.39 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [info](#)

XML (Extensible Markup Language) processing can incur significant runtime overhead in XML-based service application servers. This paper proposes a novel mechanism for efficiently processing simple document as a byte sequence, the XML parser proposed in this paper normally avoids syntactic analysis with previously processed ones, reusing those results. Our parser is adaptive since it partially processes

**Keywords:** SAX, XML parsers, automata

10 An XML model for small business e-commerce

Jim Sims, Rahul Tikekar

November 2000 **Journal of Computing Sciences in Colleges**, Volume 16 Issue 2**Publisher:** Consortium for Computing Sciences in Colleges , Consortium for Computing Sciences in Colleges

Full text available: pdf(74.30 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [info](#)

This paper proposes a model based on the eXtensible Markup Language (XML) for small business grew from the authors' and other Southern Oregon University students' experiences working with small businesses wishing to jump onto the e-commerce bandwagon. The model stems from an analysis of needs of companies who come to small colleges for advice and talent in e-commerce ventures and

11 CDuce: an XML-centric general-purpose language

Véronique Benzaken, Giuseppe Castagna, Alain Frisch

August 2003

**ACM SIGPLAN Notices , Proceedings of the eighth ACM SIGPLAN international programming ICFP '03**, Volume 38 Issue 9**Publisher:** ACM Press

Full text available: pdf(242.16 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [info](#)

We present the functional language CDuce, discuss some design issues, and show its adequacy for features of CDuce are a powerful pattern matching, first class functions, overloaded functions, a pairs, records, intersections, unions, differences), precise type inference for patterns and error handling types as sets of values. We also outline some important implementation issues ...

**Keywords:** CDuce, XML, XML-processing, type systems

12 Understanding users II: A qualitative assessment of the efficacy of UML diagrams as a formal program understanding



Scott Tilley, Shihong Huang

October 2003

**Proceedings of the 21st annual international conference on Documentatior****Publisher:** ACM PressFull text available: [pdf\(274.99 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [cit](#)

Graphical documentation is often characterized as an effective aid in program understanding. Ho types of graphical documentation are most suitable for which types of program understanding ta The Unified Modeling Language (UML) is the de facto standard for modeling modern software ap experiment to assess the qualitative efficacy of UML diagrams in aiding program understanding.

**Keywords:** Unified Modeling Language (UML), assessment, graphical documentation, program-u

### 13 Data integration and sharing I: Capturing both types and constraints in data integration



Michael Benedikt, Chee-Yong Chan, Wenfei Fan, Juliana Freire, Rajeev Rastogi

June 2003

**Proceedings of the 2003 ACM SIGMOD international conference on Manage****Publisher:** ACM PressFull text available: [pdf\(690.62 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [cit](#)

We propose a framework for integrating data from multiple relational sources into an XML docum satisfies predefined XML constraints. The framework is based on a specification language, AIG, tl types with semantic attributes (inherited and synthesized, inspired by the corresponding notions these attributes via parameterized SQL queries over multiple data sources, and (3) inc ...

### 14 From UML to LQN by XML algebra-based model transformations



Gordon P. Gu, Dorina C. Petriu

July 2005

**Proceedings of the 5th international workshop on Software and performan****Publisher:** ACM PressFull text available: [pdf\(231.43 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [inc](#)

The change of focus from code to models promoted by OMG's Model Driven Development raises characteristics of UML models. such as performance, reliability, scalability, security, etc. Many m have been developed over the years for the analysis of different non-functional characteristics. T methods for UML models, but to bridge the gap between UML-based software developmen ...

**Keywords:** LQN, UML, XMI, XML, automatic model building, model transformations, performanc

### 15 Making use of document standards and models: An environment for maintaining computati



Dongxi Liu, Zhenjiang Hu, Masato Takeichi

November 2005

**Proceedings of the 2005 ACM symposium on Document engineering DocEn****Publisher:** ACM PressFull text available: [pdf\(236.83 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [inc](#)

In the domain of XML authoring, there have been many tools to help users to edit XML documen complex documents by using such technologies as syntax-directed or presentation-oriented editi contains data with some computation dependency among them, these tools cannot free users fr dependency relationship. By computation dependency, we mean that some data are gotten by cc

**Keywords:** XML, computation dependency, functional programming, lazy evaluation, programm

### 16 Database principles: The XML typechecking problem



Dan Suciu

March 2002

**ACM SIGMOD Record**, Volume 31 Issue 1**Publisher:** ACM Press

Full text available:  [pdf\(504.57 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [cit](#)

When an XML document conforms to a given type (e.g. a DTD or an XML Schema type) it is called a valid XML document. The process of checking whether a document is valid is called the validation problem, and is typically performed by a parser (hence, performed right after parsing, by the same program module. In practice however XML document validation is often performed by a separate program: checking whether all XML documents generated by the program are valid ...

**17 Generating spreadsheet-like tools from strong attribute grammars**

João Saraiva, Doaitse Swierstra

September 2003 **Proceedings of the second international conference on Generative programming****Publisher:** Springer-Verlag New York, Inc.Full text available:  [pdf\(161.50 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [int](#)

This paper presents techniques for the formal specification and efficient incremental implementation of spreadsheets. Spreadsheets are specified by strong attribute grammars. In this style of attribute grammar programming, the spreadsheet is expressed within the attribute grammar formalism. Well-known attribute grammar techniques for example, ordered scheduling algorithms can be used to statically guarantee termination of the evaluation.

**18 Document querying and transformation: Lazy XSL transformations**

Steffen Schott, Markus L. Noga

November 2003 **Proceedings of the 2003 ACM symposium on Document engineering****Publisher:** ACM PressFull text available:  [pdf\(335.83 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [int](#)

We introduce a lazy XSLT interpreter that provides random access to the transformation result. Transformation sequences. Nodes of the result tree are computed only upon initial access. As the output coverage propagates backwards through the pipeline. In comparative measurements with other approaches, our approach is on par for complete coverage and excels as coverage becomes sparser. In contrast to other approaches, our approach is on par for complete coverage and excels as coverage becomes sparser. In contrast to other approaches, our approach is on par for complete coverage and excels as coverage becomes sparser.

**19 Integrating software productivity tools into Eclipse**

Will Robinson, Ben D'Angelo

October 2003 **Proceedings of the 2003 OOPSLA workshop on eclipse technology eXchange****Publisher:** ACM PressFull text available:  [pdf\(333.64 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#)

This paper presents a framework designed to ease the process of building GUIs for software productivity tools. The framework provides a general system with which external tools may display graphs and trees within Eclipse. In addition, our framework can retrieve information from Eclipse, passing it back to the user's external tool. Combining these pieces of functionality gives users an intuitive and powerful interface ...

**20 Session 6: Customizable description and dynamic discovery for web services**

Wooyoung Kim, Alan H. Karp




May 2004 **Proceedings of the 5th ACM conference on Electronic commerce****Publisher:** ACM PressFull text available:  [pdf\(188.80 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [int](#)

We present a framework for developing ontologies suitable for a dynamic environment, such as a commercial system for resource discovery. This framework recognizes the importance of standardization and doesn't disrupt those adhering to the standards. The framework is based on the notion of discovery security. The specific ontology we use in the system includes some salient features, ...

**Keywords:** ontologies, service description and discovery, web services



The ACM Portal is published by the Association for Computing Machinery. Copyright  
[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)